



TUGAS AKHIR - IF184802

PERANCANGAN DAN IMPLEMENTASI SISTEM LAYANAN RUMAH SAKIT TERINTEGRASI

ELVEGA DEWANGGA RACHMATULLAH
NRP 05111640000149

Dosen Pembimbing I
Ir. Muchammad Husni, M.Kom.

Dosen Pembimbing II
Tohari Ahmad, S.Kom, M.IT., Ph.D.

Departemen Teknik Informatika
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember
Surabaya 2020

(Halaman ini sengaja dikosongkan)



TUGAS AKHIR - IF184802

PERANCANGAN DAN IMPLEMENTASI SISTEM LAYANAN RUMAH SAKIT TERINTEGRASI

ELVEGA DEWANGGA RACHMATULLAH
NRP 05111640000149

Dosen Pembimbing I
Ir. Muchammad Husni, M.Kom.

Dosen Pembimbing II
Tohari Ahmad, S.Kom, M.IT., Ph.D.

Departemen Teknik Informatika
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember
Surabaya 2020

(Halaman ini sengaja dikosongkan



UNDERGRADUATE THESIS - IF184802

DESIGN AND IMPLEMENTATION OF INTELLIGENT HOSPITAL SERVICES

**ELVEGA DEWANGGA RACHMATULLAH
NRP 05111640000149**

**First Advisor
Ir. Muchammad Husni, M.Kom.**

**Second Advisor
Tohari Ahmad, S.Kom, M.IT.,Ph.D.**

**Department of Informatics Engineering
Faculty of Intelligent Electrical and Informatics Technology
Sepuluh Nopember Institute of Technology
Surabaya 2020**

(Halaman ini sengaja dikosongkan)

LEMBAR PENGESAHAN

PERANCANGAN DAN IMPLEMENTASI SISTEM LAYANAN RUMAH SAKIT TERINTEGRASI

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Bidang Studi Arsitektur dan Jaringan Komputer
Program Studi S-1 Informatika
Departemen Informatika
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember


Oleh:

Elvega Dewangga Rachmatullah
NRP: 051116 40000 149

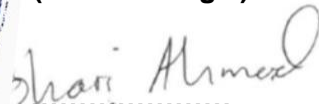
Disetujui oleh Dosen Pembimbing Tugas Akhir:

Ir. Muchammad Husni, M.Kom.
(NIP. 196002211984031001)




.....
(Pembimbing 1)

Tohari Ahmad, S.Kom, M.IT., Ph.D.
(NIP. 197505252004121002)


.....
(Pembimbing 2)

SURABAYA
JUNI 2020

v

(Halaman ini sengaja dikosongkan)

PERANCANGAN DAN IMPLEMENTASI SISTEM RUMAH SAKIT TERINTEGRASI

Nama Mahasiswa : Elvega Dewangga Rachmatullah
NRP : 05111640000149
Departemen : Teknik Informatika FTEIC ITS
Dosen Pembimbing 1 : Ir. Muchammad Husni, M.Kom.
Dosen Pembimbing 2 : Tohari Ahmad, S.Kom,
M.IT.,Ph.D.

Abstrak

Permasalahan dalam Tugas Akhir ini diambil dari dunia nyata. Secara umum persoalan yang akan diselesaikan pada penelitian Tugas Akhir ini dapat direpresentasikan sebagai persoalan menghindari antrian pemeriksaan dan membantu pasien dalam proses transaksi. Metode penyelesaian yang digunakan adalah pemanfaatan teknologi IoT.

Proses registrasi pemeriksaan akan dilakukan secara online melalui browser yang nantinya akan mendapatkan qr code untuk proses verifikasi. Terdapat juga fitur monitoring ketersediaan kamar secara realtime melalui database dipadukan dengan tombol yang ada di setiap kamar. Serta fitur pengambilan obat menggunakan proses verifikasi online.

Rumah Sakit Terintegrasi diimplementasikan dengan menggunakan Bahasa pemrograman C yang dicompile melalui Arduino IDE dan database MySQL. Dari uji coba yang telah dilakukan, didapatkan kesimpulan bahwa alat telah berhasil diimplementasikan dan berfungsi dengan baik serta dapat dilakukan website monitoring untuk pengecekan presensi dan absensi menggunakan teknologi IoT telah sesuai untuk memecahkan persoalan yang ada.

Kata kunci: *Teknologi IoT, MySQL, Arduino IDE, Bahasa Pemrograman C.*

DESIGN AND IMPLEMENTATION OF SMART HOSPITAL PROTTOTYPE

Student's Name : Elvega Dewangga Rachmatullah
Student's ID : 05111540000149
Department : Teknik Informatika FTEIC-ITS
First Advisor : Ir. Muchammad Husni, M.Kom.
Second Advisor : Tohari Ahmad, S.Kom, M.IT.,Ph.D.

Abstract

The problems in this Final Project are taken from real world. In general, the problems that will be resolved in this Final Project research can be represented as a problem of increasing safety and saving electricity in a classroom. The method that is used is the utilization of IoT technology.

Check registration process will be done online through a browser which will get a qrcode for the verification process. There is also a feature of monitoring room availability in realtime through a database integrated with the buttons in each room. As well as the drug taking feature using the online verification process.

Smart Hospital is implemented using the C programming language compiled via the Arduino IDE and MySQL database. From the trials that have been carried out, it was concluded that the tool has been successfully implemented and functioning properly and a monitoring website can be done to check the attendance using IoT technology is suitable to solve the exiting problems.

Keyword: IoT Technology, MySQL, C Programming Language, Arduino IDE.

KATA PENGANTAR

Puji syukur kepada Allah Yang Maha Esa atas segala karunia dan rahmat-Nya sehingga penulis dapat menyelesaikan tugas akhir yang berjudul

“PERANCANGAN DAN IMPLEMENTASI SISTEM LAYANAN RUMAH SAKIT TERINTEGRASI”.

Harapan dari penulis, semoga apa yang tertulis di dalam buku tugas akhir ini dapat bermanfaat bagi pengembangan ilmu pengetahuan saat ini dan ke depannya, serta dapat memberikan kontribusi yang nyata.

Dalam pelaksanaan dan pembuatan tugas akhir ini tentunya sangat banyak bantuan yang penulis terima dari berbagai pihak, tanpa mengurangi rasa hormat penulis ingin mengucapkan terima kasih sebesar-besarnya kepada:

1. Allah SWT. Dan Nabi Muhammad SAW. yang telah membimbing penulis selama hidup.
2. Keluarga penulis (Ayah, Ibu, Kakak dan keluarga penulis yang lain) yang selalu memberikan dukungan baik berupa doa, moral, dan material yang tak terhingga kepada penulis, sehingga penulis dapat menyelesaikan Tugas Akhir ini.
3. Bapak Ir. Muchammad Husni, M.Kom. dan Bapak Tohari Ahmad, S.Kom, M.IT.,Ph.D. selaku Dosen Pembimbing penulis yang telah membimbing, memberikan nasihat, dan memotivasi penulis sehingga penulis dapat menyelesaikan Tugas Akhir ini.
4. Teman-teman angkatan 2016 jurusan Teknik Informatika ITS yang telah menemani perjuangan penulis selama masa perkuliahan.

Penulis telah berusaha sebaik-baiknya dalam menyusun Tugas Akhir ini. Namun, penulis memohon maaf apabila terdapat kekurangan, kesalahan maupun kelalaian yang telah penulis lakukan. Kritik dan saran yang membangun dapat disampaikan sebagai bahan perbaikan selanjutnya. Tetaplah

berjuang karena akan ada selalu jalan untuk orang yang tidak pernah menyerah. Semoga kita semua selalu diberi kebahagiaan lahir dan batin dan kesuksesan dunia akhirat.

Surabaya, Juni 2020

Elvega Dewangga Rachmatullah

DAFTAR ISI

LEMBAR PENGESAHAN	Error! Bookmark not defined.
Abstrak	vii
Abstract	viii
KATA PENGANTAR	ix
DAFTAR ISI	xi
DAFTAR GAMBAR	xv
DAFTAR TABEL	xvii
1. BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Permasalahan	2
1.4 Tujuan	2
1.5 Manfaat	2
1.6 Metodologi	3
1.6.1 Penyusunan Proposal Tugas Akhir	3
1.6.2 Studi Literatur	3
1.6.3 Implementasi Sistem	3
1.6.4 Pengujian dan Evaluasi	4
1.6.5 Penyusunan Buku	4
1.7 Sistematika Penulisan Laporan	4
2. BAB II TINJAUAN PUSTAKA	7
2.1 WeMos D1 R1	7
2.2 Button	7
2.3 Breadboard	8
2.4 Kabel Jumper	9
2.5 Arduino IDE	9
2.6 <i>Library</i>	10
2.7 XAMPP	11

2.8	Laravel 5	12
2.9	Vscode	12
3. BAB III PERANCANGAN.....		13
3.1	Perancangan Sistem	13
3.2	Perancangan <i>Database</i>	22
3.3	Perancangan Alat	30
3.4	Perancangan Sistem <i>Monitoring</i> dan <i>Controlling</i> Rumah Sakit Terintregasi.....	31
4. BAB IV IMPLEMENTASI		35
4.1	Implementasi Database.....	35
4.1.1	Implementasi Tabel.....	35
4.1.2	Hasil Query MySQL.....	40
4.2	Implementasi Alat	40
4.2.1	Rangkaian Alat Rumah Sakit Terintregasi	41
4.2.2	Implementasi Arduino.....	42
4.3	Implementasi Sistem berbasis Web	46
4.3.1	Daftar Priksa	49
4.3.2	Monitoring Kamar	50
4.3.3	Reservasi Kamar	50
4.3.4	List Kamar Reserved	53
4.3.5	Pemberian Obat	53
4.3.6	Carts.....	54
4.3.8	Scan QR Code	56
5. BAB V UJICoba DAN EVALUASI		58
5.1	Lingkup Uji Coba	58
5.2	Sistem Rumah Sakit Terintregasi.....	58
6. BAB VI KESIMPULAN		68
6.1	Kesimpulan	68
6.2	Saran.....	68
DAFTAR PUSTAKA.....		70
LAMPIRAN		71

BIODATA PENULIS	72
------------------------------	-----------

(Halaman ini sengaja dikosongkan)

DAFTAR GAMBAR

Gambar 2.1 WeMos D1 R1	Error! Bookmark not defined.
Gambar 2.2 Button.....	Error! Bookmark not defined.
Gambar 2.3 Breadboard	Error! Bookmark not defined.
Gambar 2.4 Kabel Jumper	Error! Bookmark not defined.
Gambar 2.5 Arduino IDE.....	Error! Bookmark not defined.
Gambar 4.1 Hasil Eksekusi MySQL.....	40
Gambar 4.10 Konfigurasi database	Error! Bookmark not defined.
Gambar 5.1 Tampilan Registrasi User	59
Gambar 5.2 Tampilan List Registrasi.....	60
Gambar 5.3 Barang berhasil ditambah	60
Gambar 5.4 Halaman List kamar	61

(Halaman ini sengaja dikosongkan)

DAFTAR TABEL

Tabel 2.1 Library Arduino.....	10
Tabel 1.1 Deskripsi Tabel Database.....	22
Tabel 1.2 Detail Tabel users.....	23
Tabel 1.3 Detail Tabel doctors	24
Tabel 1.4 Detail Tabel Medicals	24
Tabel 1.5 Detail Tabel prescriptions	25
Tabel 5.1 Spesifikasi	58

(Halaman ini sengaja dikosongkan)

BAB I

PENDAHULUAN

1.1 Latar Belakang

Rumah sakit adalah sebuah institusi perawatan kesehatan profesional yang pelayanannya disediakan oleh dokter, perawat, dan tenaga ahli kesehatan lainnya. Berbeda dari Klinik Rumah sakit diharuskan memiliki layanan rawat inap bagi pasien yang memiliki penyakit serius. Sayangnya terkadang saat pasien ingin melakukan pemeriksaan kesehatan mereka harus menunggu dengan waktu yang cukup lama dalam kondisi sakit, belum lagi tidak setiap Rumah sakit dapat menyediakan Kamar bagi seluruh pasien yang harus menjalani Rawat inap sehingga pasien harus menghubungi banyak rumah sakit berbeda hanya untuk sebuah kamar.

Untuk itu, perlu adanya sistem yang dapat memudahkan Registrasi pada Suatu Rumah sakit. dengan membuat sebuah aplikasi Rumah Sakit Terintegrasi berbasis Website yang digunakan untuk Registrasi Pemeriksaan kesehatan, dan juga menampilkan nomor antrian. Lalu Setelah menjalani pemeriksaan kesehatan pasien akan mendapatkan Qr-code baru untuk mengambil obat sekaligus melakukan pembayaran secara online. Bagi pasien yang akan akan menjalani Rawat inap juga dapat melakukan pemilihan kamar secara spesifik. Dengan memadukan konsep Internet of Things (IoT), dimana Pasien dapat langsung melakukan konfirmasi nomor antrian dengan scan QR-Code yang terdapat di receptionis dan Switch yang akan digunakan Setiap Ruangan guna melakukan pemantauan ketersediaan kamar secara Real-time[3]. Dengan adanya sistem tersebut, diharap nantinya membantu mengurangi lama proses Registrasi.

1.2 Rumusan Masalah

Rumusan masalah yang diangkat dalam tugas akhir ini dapat dipaparkan sebagai berikut:

1. Bagaimana cara Menghindari antri pada proses registrasi di rumah sakit ?
2. Bagaimana cara mengetahui ketersediaan kamar untuk rawat inap pada rumah sakit di suatu kota ?
3. Bagaimana cara mempersingkat waktu pelayanan resep obat untuk pasien?
4. Bagaimana konfirmasi transaksi dengan scan QR-Code?

1.3 Batasan Permasalahan

Permasalahan pada pembuatan Tugas Akhir ini memiliki beberapa batasan. Batasan-batasannya adalah sebagai berikut:

1. Sistem yang dibangun berupa prototype alat.
2. Sistem pengaturan berbasis Aplikasi Website.
3. Untuk mengakses sistem ini memerlukan akses internet..
4. Layanan yang ditawarkan berupa pemeriksaan, rawat inap, resep obat, dokter

1.4 Tujuan

Tujuan dari pembuatan Tugas Akhir ini adalah untuk merancang Sistem Rumah Sakit Terintegrasi.

1.5 Manfaat

Manfaat dari Tugas Akhir ini diharapkan dapat membantu Sistem Rumah sakit mengurangi lama proses Registrasi, dan mempermudah pasien dalam proses Pemeriksaan Kesehatan maupun rawat inap di rumah sakit.

1.6 Metodologi

Pembuatan Tugas Akhir ini dilakukan dengan menggunakan metodologi sebagai berikut:

1.6.1. Penyusunan Proposal Tugas Akhir

Tahapan awal dari Tugas Akhir ini adalah penyusunan Proposal Tugas Akhir. Proposal Tugas Akhir berisi pendahuluan, deskripsi dan gagasan metode-metode yang dibuat dalam Tugas Akhir ini. Pendahuluan ini terdiri atas hal yang menjadi latar belakang diajukannya Tugas Akhir, rumusan masalah yang diangkat, batasan masalah untuk Tugas Akhir, dan manfaat dari hasil pembuatan Tugas Akhir ini. Selain itu dijabarkan pula tinjauan pustaka yang digunakan sebagai referensi pendukung pembuatan Tugas Akhir.

1.6.2. Studi Literatur

Studi literatur yang dipakai adalah dengan pengumpulan informasi mengenai bahan-bahan yang digunakan dalam pembuatan prototipe yang dijadikan sebagai referensi dalam pengerjaan Tugas Akhir. Mengumpulkan informasi mengenai IoT(*Internet of Things*), sensor, *solenoid*, *wemos* dan *arduino*. Informasi didapatkan dari buku, paper, jurnal dan materi-materi kuliah yang berhubungan dengan topik Tugas Akhir.

1.6.3. Implementasi Sistem

Implementasi yang akan dilakukan yaitu prototipe perancangan sistem berdasarkan studi literatur dan pengumpulan informasi yang telah dilakukan. Tahap ini suatu bentuk awal perancangan yang akan diimplementasikan. Yang dibuat adalah merupakan prototipe versi awal perancangan system Rumah Sakit Terintegrasi.

1.6.4. Pengujian dan Evaluasi

Pengujian dilakukan melalui dua cara yaitu:

1. Pengujian *hardware*
Pengujian *hardware* adalah pengujian yang berfokus pada spesifikasi fungsional dari perangkat keras. Pengujian ini dilakukan untuk menguji apakah sistem Rumah Sakit Terintregasi berjalan atau tidak.
2. Pengujian *software*
Pengujian *software* adalah pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak. Pengujian ini dilakukan sistem monitoring Rumah Sakit Terintregasi berjalan atau tidak.

1.6.5. Penyusunan Buku

Pada tahap ini dilakukan penyusunan buku sebagai dokumentasi dari pelaksanaan tugas akhir yang mencakup seluruh konsep, teori, implementasi, serta hasil yang telah dikerjakan.

1.7 Sistematika Penulisan Laporan

Sistematika penulisan laporan tugas akhir adalah sebagai berikut:

Bab I Pendahuluan

Bab ini berisi penjelasan latar belakang, rumusan masalah, batasan masalah dan tujuan pembuatan Tugas Akhir. Selain itu, metodologi pengerjaan dan sistematika penulisan laporan Tugas Akhir juga dijelaskan di dalamnya.

Bab II Tinjauan Pustaka

Pada bagian ini berisi kajian teori atau penjelasan dari metode, algoritma, *library*, dan *tools* yang digunakan dalam penyusunan Tugas Akhir ini.

Bab III Perancangan Sistem

Bab ini berisi penjelasan tentang rancangan dari sistem yang akan dibangun.

Bab IV Implementasi

Pada bagian ini menjelaskan implementasi yang berbentuk kode program dari proses modifikasi, proses pengujian, serta hardware dan maket.

Bab V Pengujian dan Evaluasi

Bab ini berisi penjelasan mengenai data hasil percobaan dan pembahasan mengenai hasil percobaan yang telah dilakukan.

Bab VI Kesimpulan dan Saran

Bab ini merupakan bab terakhir yang menjelaskan kesimpulan dari hasil ujicoba yang dilakukan dan saran untuk pengembangan perangkat lunak ke depannya.

(Halaman ini sengaja dikosongkan)

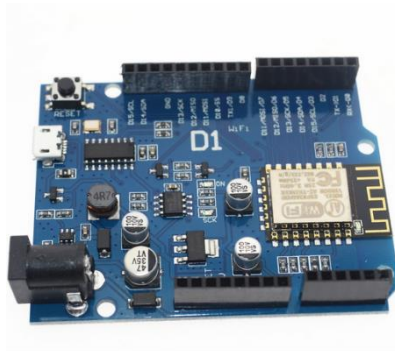
BAB II

TINJAUAN PUSTAKA

Bab ini berisi penjelasan mengenai teori-teori dasar yang berkaitan dengan pengimplementasian perangkat lunak dan perangkat keras serta penunjangnya. Penjelasan ini bertujuan untuk memberikan gambaran secara umum terhadap *Internet of Things*, tools, serta definisi yang digunakan dalam pembuatan Tugas Akhir.

1.1 WeMos D1 R1

WeMos D1 merupakan *module development board* yang berbasis *WiFi* dari ESP8266 yang dimana dapat diprogram menggunakan *software* IDE Arduino, perangkat inilah yang nantinya akan menghubungkan perangkat dengan *internet* melalui *WiFi*.



Gambar 2.1 Wemos

1.2 Button

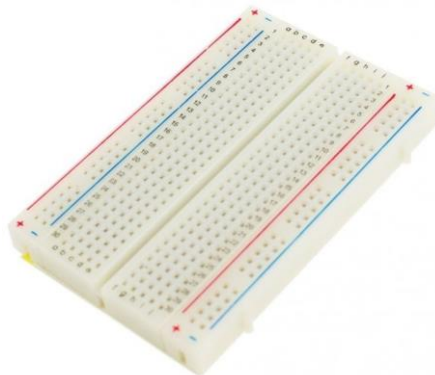
Button adalah perangkat keras universal yang digunakan hampir disetiap elektronik yang kita miliki, *Button* berfungsi untuk Mengirimkan sinyal status ke *module development board*



Gambar 2.2 Button

1.3 Breadboard

Breadboard adalah alat untuk dapat dengan cepat menguji sirkuit prototipe atau mengaitkan sebuah eksperimen cepat. Breadboarding adalah langkah penting dalam menguji ide atau mempelajari cara kerja sesuatu



Gambar 2.3 Breadboard

1.4 Kabel Jumper

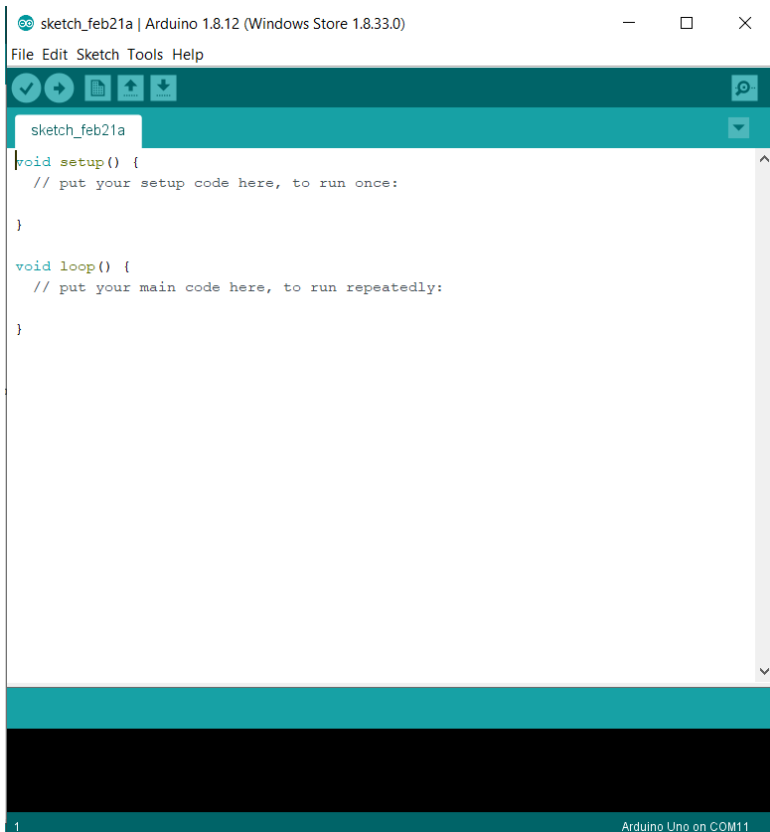
Kabel jumper adalah kabel yang digunakan untuk menghubungkan Arduino dengan sensor atau modul lainnya, terdapat tiga jenis kabel jumper, Female to Male, Female to Female, Male to Male, dimana masing masing memiliki fungsi yang sama hanya berbeda pada ujung kabelnya saja.



Gambar 2.4 Kabel Jumper

1.5 Arduino IDE

Arduino Integrated Development Environment (IDE) merupakan editor teks untuk menulis kode, area pesan, konsol teks dan memiliki toolbar dengan tombol untuk fungsi umum dan serangkaian menu. Arduino IDE terhubung ke perangkat keras Arduino dan Genuino untuk mengunggah program dan dapat berkomunikasi dengan meraka. Program yang ditulis menggunakan Arduino Software (IDE) disebut sketsa. Sketsa ini ditulis dalam editor teks dan disimpan dengan ekstensi file .ino. Editor memiliki fitur untuk memotong / menempel dan mencari / mengganti teks. Area pesan memberi umpan balik saat menyimpan dan mengekspor dan menampilkan kesalahan. Konsol menampilkan output teks oleh Arduino Software (IDE), termasuk pesan kesalahan dan informasi lainnya yang lengkap.



Gambar 2.5 Arduino IDE

1.6 Library

Dalam pembuatan Tugas Akhir ini digunakan beberapa *library* untuk mendukung berjalannya pemograman pada Arduino, beberapa *library* yang digunakan adalah:

Tabel 2.1 Library Arduino

Nama Library	Deskripsi
MFRC522.h	<i>Library</i> yang digunakan untuk membaca dan menulis berbagai jenis kartu yang memiliki <i>Radio-Frequency IDentification</i> (RFID) pada Arduino dengan menggunakan pembaca berbasis RC522 yang terhubung melalui antarmuka <i>Serial Peripheral Interface</i> (SPI).
Servo.h	<i>Library</i> yang memungkinkan Arduino untuk menggerakkan <i>Servo Motor</i> .
MySQL_Connection.h MySQL_Cursor.h	<i>Library</i> yang memungkinkan Arduino untuk berinteraksi secara langsung dengan <i>database</i> menggunakan <i>internet</i> .
ESP8266WiFi.h	<i>Library</i> yang digunakan untuk menghubungkan ESP8266 pada Arduino dengan <i>internet</i> .
ESP8266HTTPClient.h WiFiClient.h	<i>Library</i> yang memungkinkan untuk ESP8266 pada Arduino mengakses sebuah <i>Hypertext Transfer Protocol</i> (HTTP).

1.7 XAMPP

XAMPP adalah sebuah paket perangkat lunak (software) komputer yang sistem penamaannya diambil dari akronim kata Apache, MySQL (dulu) / MariaDB (sekarang), PHP, dan Perl. Sementara imbuhan huruf “X” yang terdapat pada awal kata

berasal dari istilah *cross platform* sebagai simbol bahwa aplikasi ini bisa dijalankan di empat sistem operasi berbeda, seperti OS Linux, OS Windows, Mac OS, dan juga Solaris. XAMPP pada Tugas Akhir ini akan digunakan sebagai *WebServer* serta penggunaan *database* MySQL yang telah tersedia pada XAMPP.

1.8 Laravel 5

Laravel 5 adalah sebuah *Framework* berbasis PHP yang digunakan pada pembuatan *website* untuk melakukan *Monitoring* Rumah Sakit Terintregasi.

1.9 Vscode

Vscode adalah *Integrated Development Environment* (IDE) yang dapat digunakan dalam melakukan pengembangan aplikasi berbasis website. Selain sebagai IDE, Visual Studio code menawarkan lebih banyak fitur dan *extension* yang meningkatkan produktivitas saat membangun aplikasi berbasis website.

BAB III

PERANCANGAN

Pada bagian ini akan dijelaskan rancangan *hardware* serta sistem yang akan dibangun oleh penulis pada Tugas Akhir ini.

3.1 Perancangan Sistem

Perancangan dan Implementasi Prototipe Sistem Rumah Sakit Terintregasi ini adalah sebuah *prototype* yang didesain untuk mengurangi membantu sistem rumah sakit menghindari lama antrian, dan tidak mempermudah pasien memilih kamar rawat inap berbasis *microcontroller* WeMos. Untuk dapat berfungsi, alat ini harus terhubung dengan koneksi internet melalui *Wi-Fi*, yang nantinya akan mencatat dan memvalidasi pasien yang melakukan *Scan QR-Code*. Untuk melakukan implementasi *prototype*, dibutuhkan beberapa alat sebagai berikut :

1. Aplikasi Rumah Sakit Terintregasi

Aplikasi Rumah Sakit Terintregasi ini adalah sebuah aplikasi berbasis website yang memiliki beberapa fungsi, yaitu:

- a. Melakukan registrasi pemeriksaan secara online
- b. Melakukan konfirmasi registrasi pemeriksaan dengan *scan QR-Code*.
- c. Melihat list kamar pada setiap rumah sakit
- d. Melakukan konfirmasi reservasi kamar dengan *scan QR-Code*.
- e. Melihat list pemberian resep obat secara online
- f. Melakukan konfirmasi pengambilan resep obat dengan *scan QR-Code*.
- g. Melihat list dokter pada setiap rumah sakit.

2. Aplikasi Rumah Sakit Terintregasi (admin)

Aplikasi Rumah Sakit Terintregasi (admin) adalah sistem berbasis *website* yang dibuat untuk admin atau petugas Rumah Sakit dalam melakukan *monitoring* dan

controlling terhadap Rumah Sakit Terintegrasi, dimana sistem monitor dapat melakukan :

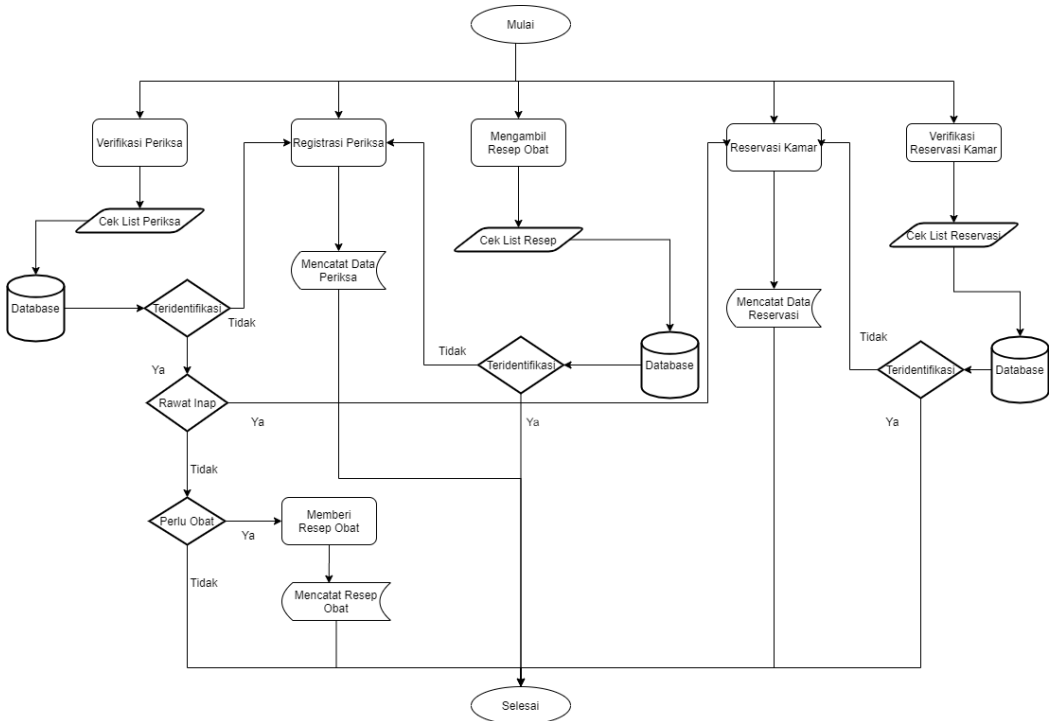
- a. Melihat list pasien periksa pada setiap dokter.
- b. Melakukan reservasi kamar secara online
- c. Melakukan monitoring kamar secara online
- d. Melakukan Pemberian resep obat secara online
- e. Melihat riwayat transaksi.

3. *Smart Hospital Device*

Smart Hospital Device Smart Hospital Device adalah sistem yang ada di setiap kasur pada kamar rumah sakit yang berfungsi sebagai:

- a. Mengirim sinyal ke *Database* jika kamar terisi oleh pasien

Proses Aplikasi Rumah Sakit Terintegrasi dapat dilihat pada Gambar 3.1 Flowchart Aplikasi Rumah Sakit Terintegrasi.

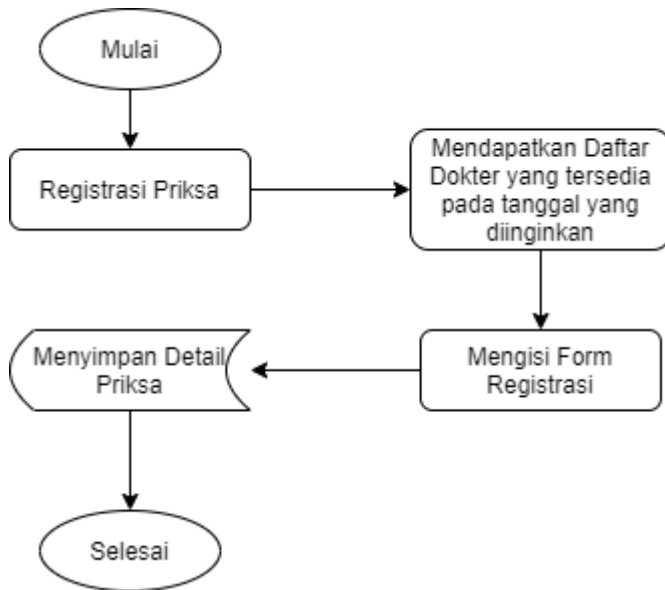


Gambar 3.1 Flowchart Aplikasi Rumah Sakit Terintegrasi

Pada *flowchart* Aplikasi Rumah Sakit Terintegrasi memiliki beberapa jenis alur, yaitu:

1. Registrasi Periksa

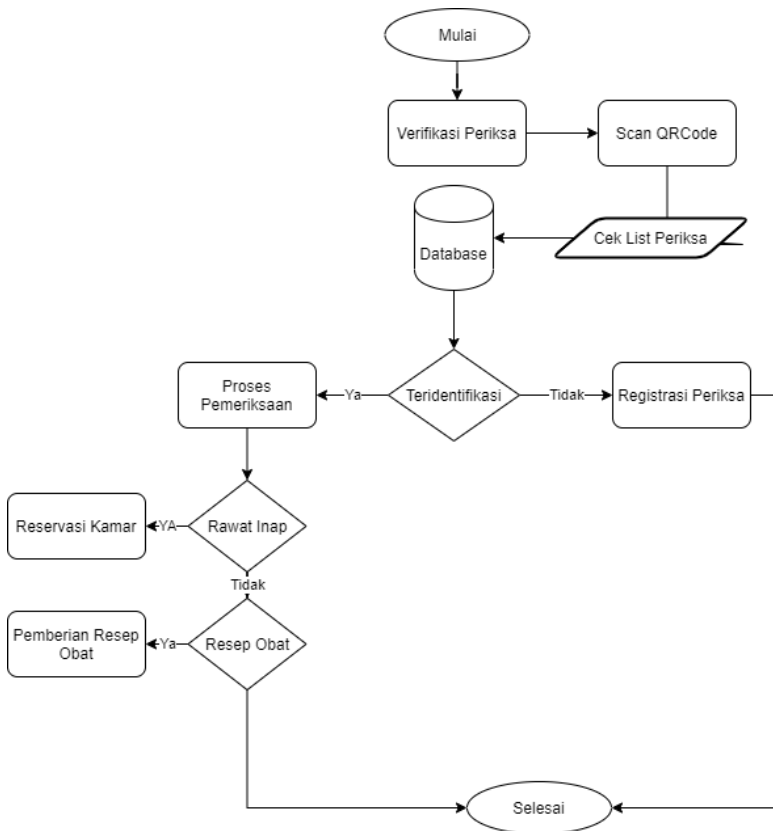
Pada proses ini pasien melakukan Registrasi Priksa melalui aplikasi. Kemudian calon pasien memasukkan data diri dan opsi dokter yang diminta aplikasi. Setelah itu data diri calon pasien akan disimpan dalam database.



Gambar 3.2 Flowchart Alur Registrasi Priksa

2. Verifikasi Priksa

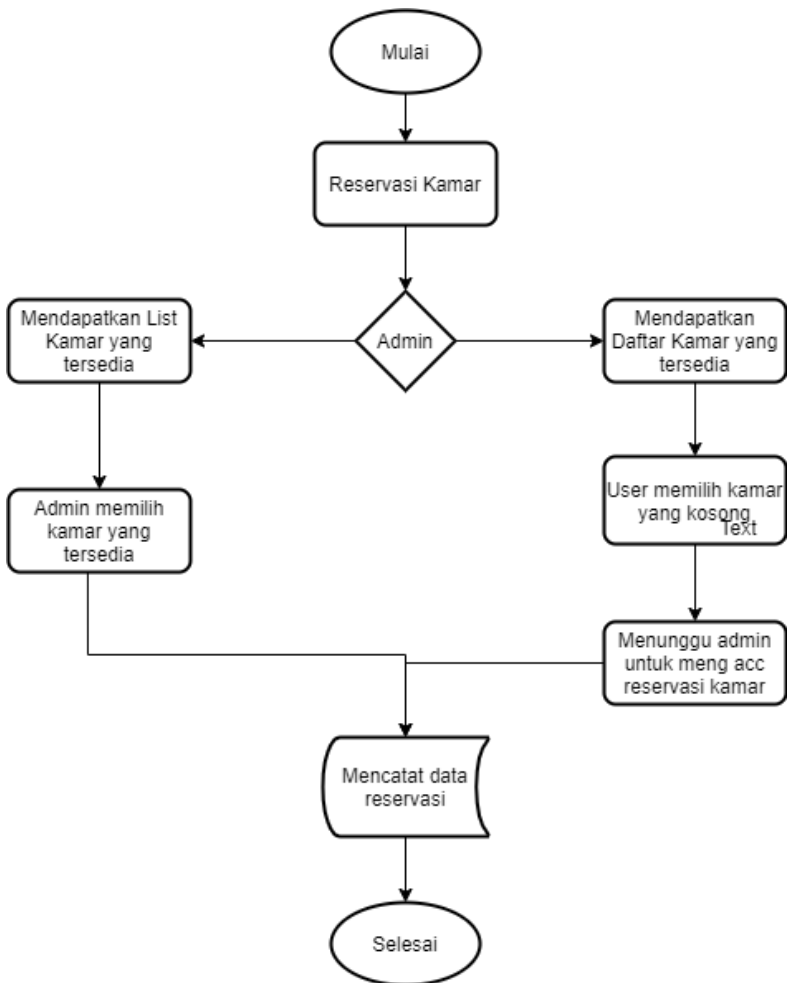
Pada proses ini pasien tiba di rumah sakit untuk melakukan Verifikasi. Apabila pasien belum melakukan registrasi, maka pasien diminta untuk melakukan flowchart alur 1 (Registrasi Priksa). Sebaliknya, setelah konfirmasi berhasil, menu Reservasi kamar dan Pemberian Resep Obat akan bisa diakses.



Gambar 3.3 Flowchart verivikasi Priksa di Rumah Sakit

3. Reservasi Kamar

Pada proses ini jika dinilai Pasien Memerlukan penanganan Rawat Inap. Petugas akan memilihkan kamar yang sesuai dengan yang diinginkan pasien Setelah itu data diri pengunjung akan disimpan dalam database:

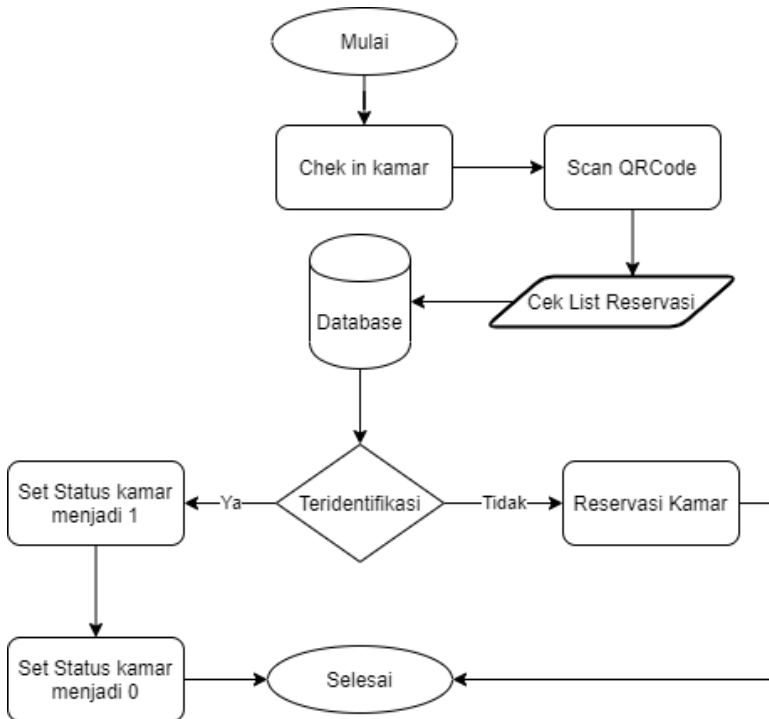


Gambar 3.4 Flowchart Reservasi Kamar di Rumah Sakit

4. Checkin kamar

Pada proses ini pasien tiba di rumah sakit untuk melakukan Verifikasi Apabila pengunjung belum melakukan registrasi, maka petugas akan membantu

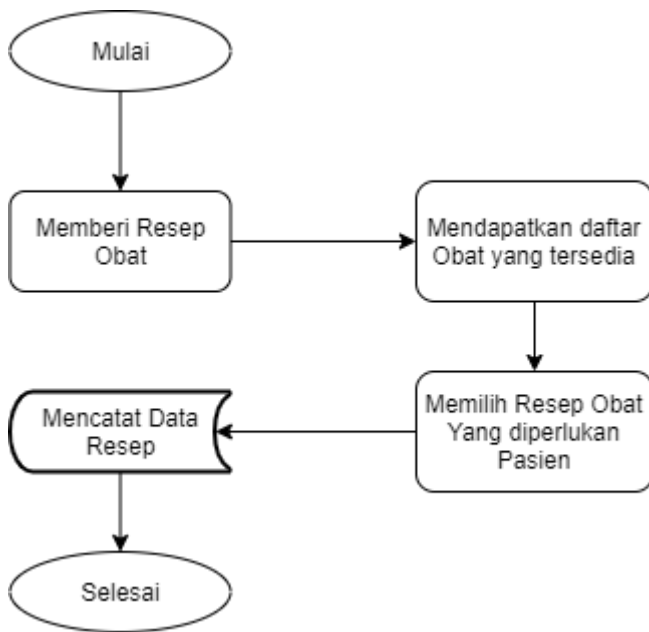
pasien untuk menjalankan flowchart alur 3(Reservasi Kamar)



Gambar 3.5 Flowchart Check In Kamar di Rumah Sakit

5. Memberi Resep Obat

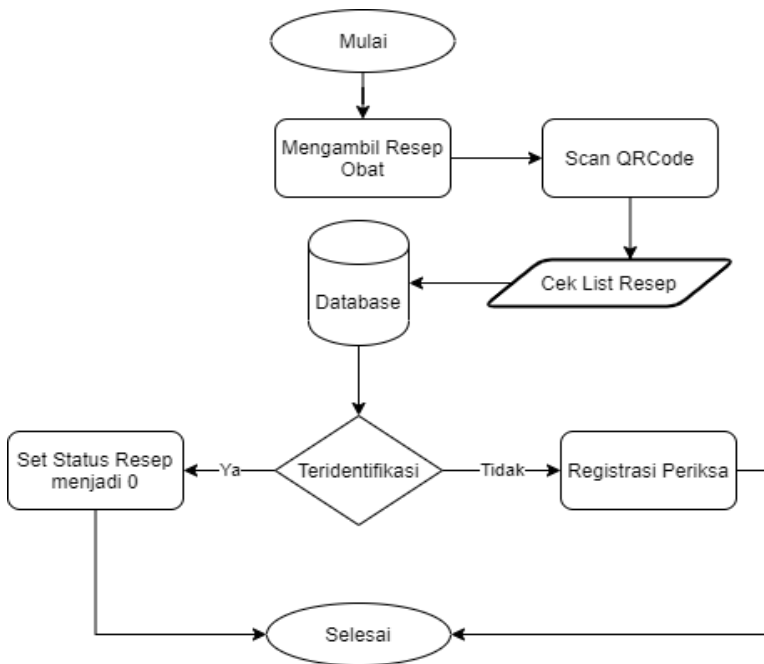
Pada proses ini jika dinilai Pasien tidak Memerlukan penanganan Rawat Inap. Dokter akan menilai apakah pasien memerlukan obat untuk membantu penyembuhan nya. Jika iya, Dokter akan memberikan resep obat yang sesuai. Setelah itu data diri pasien akan disimpan dalam database.



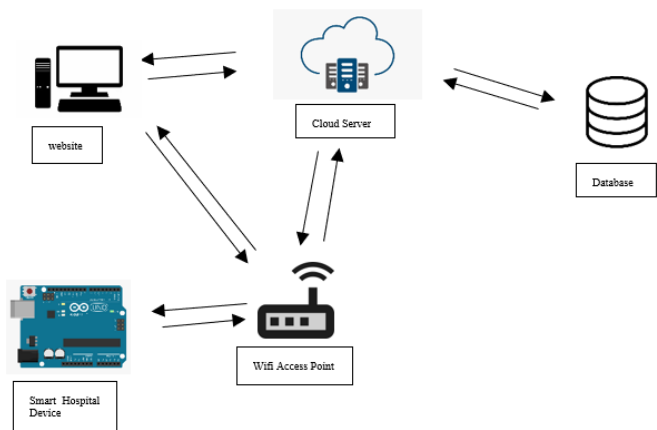
Gambar 3.6 Flowchart Memberi resep obat di Rumah Sakit

6. mengambil Resep Obat

Pada proses ini jika dinilai Pasien tidak memerlukan penanganan Rawat Inap. Dokter akan menilai apakah pasien memerlukan obat untuk membantu penyembuhan nya. Jika iya, Dokter akan memberikan resep obat yang sesuai. Setelah itu data diri pasien akan disimpan dalam database.



Gambar 3.7 Flowchart mengambil resep obat di Rumah Sakit



Gambar 3.8 Alur Interaksi Device

3.2 Perancangan *Database*

Rancangan *Database* ini digunakan sebagai media penyimpanan data oleh Web Service yang nantinya data akan berubah sesuai data yang dikirim oleh Webs Service, yang akan ditampilkan pada Aplikasi Rumah Sakit Terintegasi. *Database* akan berisi 7 tabel yaitu:

Tabel 3.1 Deskripsi Tabel Database

Nama Tabel	Deskripsi
users	Tabel yang berisi data pengguna.
doctors	Tabel yang berisi data dokter.
medicals	Tabel yang berisi data list obat
prescriptions	Tabel yang berisi data transaksi resep obat

prescriptions_details	Tabel yang berisi data detail transaksi resep obat
reservations	Tabel yang berisi data transaksi ruangan
rooms	Tabel yang berisi data ruang dalam rumah sakit
appointments	Tabel yang berisi data registrasi pendaftaran
appointment_time	Tabel yang berisi data detail pembagian waktu registrasi pendaftaran
carts	Tabel yang berisi data sementara dari transaksi resep obat

Penjelasan mengenai rincian tabel pada database adalah sebagai berikut:

Tabel 3.2 Detail Tabel users

users		
Atribut	DataType	Deskripsi
id	Bigint(20)	Atribut <i>primary key</i> pada table ini yang berfungsi sebagai ID <i>primary</i> pada tabel.
name	Varchar(255)	Atribut yang berisi nama dari user.
email	Varchar(255)	Atribut yang berisi email dari user.
nik	Varchar(255)	Atribut yang berisi nik dari user.
password	Varchar(255)	Atribut yang berisi password dari user

rules	Int(11)	Atribut yang membedakan antara admin dan user biasa.
doctor_id	Bigint(20)	Atribut <i>foreign key</i> yang berisi dokter_id dari dokter.

Tabel 3.3 Detail Tabel doctors

doctors		
Atribut	Data Type	Deskripsi
id	Bigint (20)	Atribut <i>primary key</i> pada table ini yang berfungsi sebagai ID <i>primary</i> pada tabel.
nama	Varchar(255)	Atribut yang berisi nama dari dokter.
specialis	Varchar(255)	Atribut yang berisi spesialis dari dokter.
lokasi	Varchar(255)	Atribut yang berisi lokasi dari dokter.

Tabel 3.4 Detail Tabel Medicals

medicals		
Atribut	Data Type	Deskripsi
id	Bigint(11)	Atribut <i>primary key</i> pada table ini yang berfungsi sebagai ID <i>primary</i> pada tabel.

jenis	Varchar(255)	Atribut yang berisi nama dari obat.
bentuk	Varchar(255)	Atribut yang berisi bentuk dari obat.
nama	Varchar(255)	Atribut yang berisi nama dari obat.
harga	Int(11)	Atribut yang berisi harga dari obat.

Tabel 3.5 Detail Tabel prescriptions

prescriptions		
Atribut	DataType	Deskripsi
id	Integer (11)	Atribut <i>primary key</i> pada table ini yang berfungsi sebagai ID <i>primary</i> pada tabel.
Appointment_id	Integer (11)	Atribut <i>foreign key</i> pada tabel yang didapat dari tabel appointment
total	Integer (11)	Atribut yang berisi total harga dari resep obat.
status	Integer (11)	Atribut yang berisi status dari resep obat.
ambil	Datetime	Atribut yang berisi waktu diambilnya obat.

Tabel 3.6 Detail Tabel prescriptions_details

prescriptions_details		
Atribut	DataType	Deskripsi
t_id	Integer (11)	Atribut <i>primary key</i> pada table ini yang berfungsi sebagai ID <i>primary</i> pada tabel

		ht_log_transaksi yang berfungsi sebagai penyimpanan riwayat transaksi.
u_id	Integer (11)	Atribut <i>foreign key</i> pada tabel yang didapat dari tabel ht_user.

Tabel 3.7 Detail Tabel reservation

reservations		
Atribut	Data Type	Deskripsi
id	Bigint(20)	Atribut <i>primary key</i> pada table ini yang berfungsi sebagai ID <i>primary</i> pada tabel.
nama	Varchar(255)	Atribut yang berisi nama pasien dari list pemesanan rawat inap.
telp	Varchar(255)	Atribut yang berisi Telp keluarga pasien yang bisa dihubungi
email	Varchar(255)	Atribut yang berisi Email keluarga pasien yang bisa dihubungi
status	Integer (11)	Atribut yang berisi Status apakah pasien sudah melakukan konfirmasi scan qrcode atau belum

user_id	Bigint(20)	Atribut <i>foreign key</i> yang berisi id dari user.
room_id	Bigint(20)	Atribut <i>foreign key</i> yang berisi id dari room.

Tabel 3.8 Detail Tabel rooms

rooms		
Atribut	DataType	Deskripsi
id	Bigint(20)	Atribut <i>primary key</i> pada table ini yang berfungsi sebagai ID <i>primary</i> pada tabel.
nomor	Varchar(255)	Atribut yang berisi nomor dari kamar.
kelas	Varchar(255)	Atribut yang berisi kelas dari kamar.
lokasi	Varchar(255)	Atribut yang berisi lokasi dari kamar.
status	Varchar(255)	Atribut yang berisi status dari kamar

Tabel 3.9 Detail Tabel appointments

appointments		
Atribut	DataType	Deskripsi
id	Integer (11)	Atribut <i>primary key</i> pada table ini yang berfungsi sebagai ID <i>primary</i> pada tabel.

segment_time	Enum('pagi','siang')	Atribut yang berisi pilihan waktu dari user
urutan	Integer(11)	Atribut yang berisi urutan pendaftaran dari user
tanggal	date	Atribut yang berisi tanggal registrasi priksa yang diinginkan user
lokasi	Varchar(255)	Atribut yang berisi Rumah sakit untuk priksa yang diinginkan user
nama	Varchar(255)	Atribut yang berisi nama pasien
telp	Varchar(255)	Atribut yang berisi Telp keluarga pasien yang bisa dihubungi
email	Varchar(255)	Atribut yang berisi Email keluarga pasien yang bisa dihubungi
status	Integer(11)	Atribut yang berisi Status apakah pasien sudah melakukan

		konfirmasi scan qrcode atau belum
user_id	Bigint(20)	Atribut <i>foreign key</i> yang berisi id dari user.
doctor_id	Bigint(20)	Atribut <i>foreign key</i> yang berisi id dari doctor

Tabel 3.10 Detail Tabel *appointments_time*

appointments_time		
Atribut	DataType	Deskripsi
id	Bigint(20)	Atribut <i>primary key</i> pada table ini yang berfungsi sebagai ID <i>primary</i> pada tabel.
urutan	Integer(11)	Atribut yang berisi urutan pendaftaran dari user
appointment_id	Bigint(20)	Atribut <i>foreign key</i> yang berisi id dari appointment
doctor_id	Bigint(20)	Atribut <i>foreign key</i> yang berisi id dari doctor
date	Date	Atribut yang berisi tanggal registrasi priksa yang diinginkan user

jam_layanan	Varchar(255)	Atribut yang berisi perkiraan giliran priksa
-------------	--------------	--

Tabel 3.11 Detail Tabel carts

carts		
Atribut	DataType	Deskripsi
user_id	Bigint(20)	Atribut <i>foreign key</i> yang berisi id dari user
medical_id	Bigint(20)	Atribut <i>foreign key</i> yang berisi id dari medical
quantity	Integer(11)	Atribut yang berisi jumlah obat

3.3 Perancangan Alat

Perancangan prototipe Rumah Sakit Terintegrasi akan menggunakan beberapa komponen *hardware*, dengan WeMos D1 R1 sebagai *microcontroller*, diantaranya :

1. Button
2. Breadboard
3. Resistor
4. Kabel Jumper

Microcontroller akan dihubungkan dengan masing-masing komponen melalui 8-Channel Relay 5V dengan mengirimkan arus listrik kepada relay, dimana *microcontroller* akan mengirimkan sinyal *High* dan *Low* kepada relay.

Ketika *microcontroller* mengirimkan sinyal *Low* maka relay akan aktif dan saklar tertutup sehingga terminal NO(Normally Open) akan mendapatkan arus, sedangkan saat *microcontroller* mengirimkan sinyal *High* maka relay akan mati dan saklar akan terbuka sehingga terminal

NC(*Normally Closed*) akan mendapatkan arus dan membuka saklar. Dan membuka saklar komponen-komponen *hardware* akan terhubung dengan relay dengan beberapa ketentuan sebagai berikut:

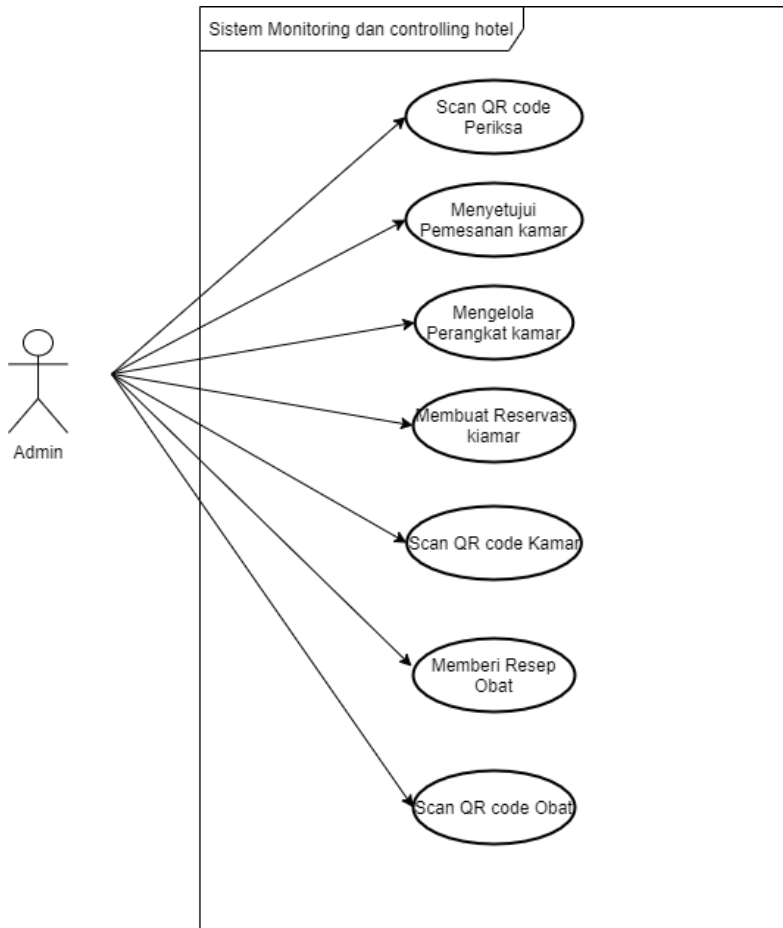
1. button terhubung pada relay IN 1 pada terminal NO.

Microcontroller yang terhubung pada 8-Channel Relay 5V akan berfungsi sebagai berikut:

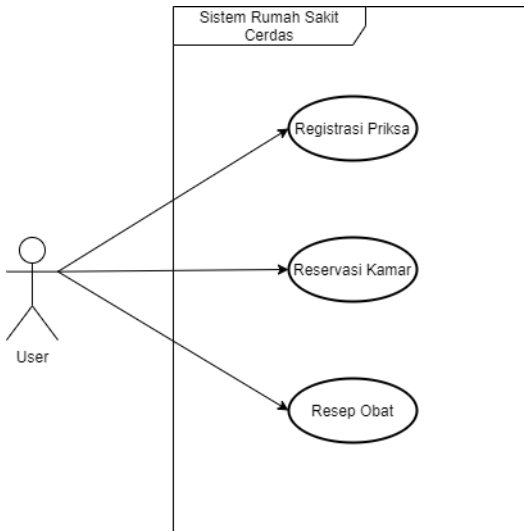
1. Mengganti status dengan mengirimkan sinyal *Low* saat pengguna admin menekan button kemudian button akan mengganti data status pada table *rooms*

3.4 Perancangan Sistem *Monitoring* dan *Controlling* Rumah Sakit Terintegrasi

Perancangan Sistem *Monitoring* dan *Controlling* Rumah Sakit Terintegrasi digambarkan melalui Use Case Diagram pada Gambar 3.7 Use Case Diagram Sistem *Monitoring* dan *Controlling* Rumah Sakit Terintegrasi.



Gambar 3.8 A Use Case Diagram Sistem Monitoring dan Controlling Rumah Sakit Terintegrasi



Gambar 3.9 A Use Case Diagram User Rumah Sakit Terintegrasikan

Sistem Monitor dibangun menggunakan *framework* Laravel dan akan digunakan oleh admin Rumah Sakit untuk melakukan *monitoring* dan *controlling* Sistem Rumah Sakit Terintegrasikan, yang dapat melakukan:

1. Registrasi Priksa
2. Pemantauan Transaksi priksa
3. Scan QRcode priksa
4. Reservasi kamar
5. Pemantauan Transaksi kamar
6. Scan QRcode kamar
7. Pemberian Resep Obat
8. Pemantauan Resep Obat
9. Scan QRcode Resep Obat

[Halaman ini sengaja dikosongkan]

BAB IV

IMPLEMENTASI

Pada bab ini akan dijelaskan tentang implementasi yang dilakukan berdasarkan perancangan pada bab sebelumnya

4.1 Implementasi Database

Pada implementasi *database* ini, penulis menggunakan *tool MySQL* untuk mengakses *database*, dan *XAMPP* sebagai penyimpanan *database MySQL* serta *Web Server*, *database* akan dibuatkan tabel seperti yang telah dijelaskan pada bab perancangan.

4.1.1 Implementasi Tabel

4.1.1.1 Implementasi Tabel user

Pada implementasi tabel user akan menjalankan *query* seperti Kode Sumber 4.1 pada *MySql*

```
1. Schema::create('users', function (Blueprint $table) {
2.     $table->bigIncrements('id');
3.     $table->string('name');
4.     $table->string('email')->unique();
5.     $table->string('nik')->unique();
6.     $table->timestamp('email_verified_at')->nullable();
7.     $table->string('password');
8.     $table->integer('rules')->default('1');
9.     $table->bigInteger('doctor_id')->unsigned()->nullable();
10.    $table->rememberToken();
11.    $table->timestamps();
12.    $table->foreign('doctor_id')->references('id')->on('doctors');
13. });
```

Kode Sumber 4.1 MySQL user

4.1.1.2 Implementasi Tabel doctors

Pada implementasi tabel doctor akan menjalankan *query* seperti Kode Sumber 4.2 pada MySql

```
1. Schema::create('doctors', function (Blueprint $table) {
2.     $table->bigIncrements('id');
3.     $table->string('nama')->nullable();
4.     $table->string('specialis')->nullable();
5.     $table->string('lokasi')->nullable();
6. });
```

Kode Sumber 4.2 MySQL doctors

4.1.1.3 Implementasi Tabel medicals

Pada implementasi tabel medicals akan menjalankan *query* seperti Kode Sumber 4.3 pada *MySql*.

```
1. Schema::create('medicals', function (Blueprint $table) {
2.     $table->bigIncrements('id');
3.     $table->string('jenis')->nullable();
4.     $table->string('bentuk')->nullable();
5.     $table->string('nama')->nullable();
6.     $table->integer('harga')->nullable();
7. });
```

Kode Sumber 4.3 MySQL medicals

4.1.1.4 Implementasi Tabel prescriptions

Pada implementasi tabel prescriptions akan menjalankan *query* seperti Kode Sumber 4.4 pada *MySql*.

```
1. Schema::create('prescriptions', function (Blueprint $table) {
2.     $table->bigIncrements('id');
3.     $table->bigInteger('appointment_id')->unsigned();
4.     $table->integer('status')->default('1');
5.     $table->integer('total');
```



```

6.   $table->dateTime('ambil');
7.   $table->timestamps();
8.   $table->foreign('appointment_id')->references('id')-
    >on('appointments');
9.   });

```

Kode Sumber 4.4 MySQL prescription

4.1.1.5 Implementasi Tabel prescriptions_details

Pada implementasi tabel prescriptions_details akan menjalankan *query* seperti Gambar 4.5 pada *MySQL*.

```

1.   Schema::table('prescriptions_details', function (Blueprint $table) {
2.       $table->bigInteger('prescription_id')->unsigned();
3.       $table->bigInteger('medical_id')->unsigned();
4.       $table->integer('quantity');
5.       $table->timestamps();
6.       $table->foreign('prescription_id')->references('id')-
    >on('prescriptions');
7.       $table->foreign('medical_id')->references('id')-
    >on('medicals');
8.   });

```

Kode Sumber 4.5 MySQL prescription_details

4.1.1.6 Implementasi Tabel reservations

Pada implementasi tabel reservations akan menjalankan *query* seperti Kode Sumber 4.6 pada *MySQL*.

```

1.   Schema::table('reservations', function (Blueprint $table) {
2.       $table->bigIncrements('id');
3.       $table->bigInteger('user_id')->unsigned();
4.       $table->bigInteger('doctor_id')->unsigned();
5.       $table->bigInteger('room_id')->unsigned();
6.       $table->timestamps();
7.       $table->foreign('user_id')->references('id')->on('users');
8.       $table->foreign('doctor_id')->references('id')-
    >on('doctors');

```

```

9.         $table->foreign('room_id')->references('id')->on('rooms');
10.    });

```

Kode Sumber 4.6 MySQL reservations

4.1.1.7 Implementasi Tabel rooms

Pada implementasi tabel rooms akan menjalankan *query* seperti Kode Sumber 4.7 pada *MySQL*.

```

1.  Schema::create('rooms', function (Blueprint $table) {
2.      $table->bigIncrements('id');
3.      $table->string('nomor')->nullable();
4.      $table->string('kelas')->nullable();
5.      $table->string('lokasi')->nullable();
6.      $table->string('status')->nullable();
7.      $table->timestamps();
8.  });

```

Kode Sumber 4.7 MySQL rooms

4.1.1.8 Implementasi Tabel appointments

Pada implementasi tabel appointments akan menjalankan *query* seperti Kode Sumber 4.8 pada *MySQL*.

```

1.  Schema::create('appointments', function (Blueprint $table) {
2.      $table->bigIncrements('id');
3.      $table->date('tanggal')->nullable();
4.      $table->string('lokasi')->nullable();
5.      $table->string('nama')->nullable();
6.      $table->string('telp')->nullable();
7.      $table->string('email')->nullable();
8.      $table->integer('status')->nullable();
9.      $table->bigInteger('user_id')->unsigned();
10.     $table->bigInteger('doctor_id')->unsigned();
11.     $table->timestamps();
12.     $table->foreign('user_id')->references('id')->on('users');
13.     $table->foreign('doctor_id')->references('id')->on('doctors');
14. });

```

Kode Sumber 4.8 MySQL appointments

4.1.1.9 Implementasi Tabel appointments_time

Pada implementasi tabel ht_log_transaksi akan menjalankan *query* seperti Kode Sumber 4.8 pada *MySQL*.

```

1. Schema::table('appointments_time', function (Blueprint
   int $table) {
2.         $table->id();
3.         $table->integer('urutan')-
   >nullable();
4.         $table->bigInteger('appointment_id')-
   >unsigned()->nullable();
5.         $table->bigInteger('doctor_id')-
   >unsigned()->nullable();
6.         $table->date('date')->nullable();
7.         $table->string('jam_layanan')-
   >nullable();$table->foreign('appointment_id')-
   >references('id')->on('appointments');
8.         $table->foreign('doctor_id')-
   >references('id')->on('doctors');
9.     });

```

Kode Sumber 4.9 MySQL appointments_time

4.1.1.10 Implementasi Tabel carts

Pada implementasi tabel carts akan menjalankan *query* seperti Kode Sumber 4.10 pada *MySQL*.

```

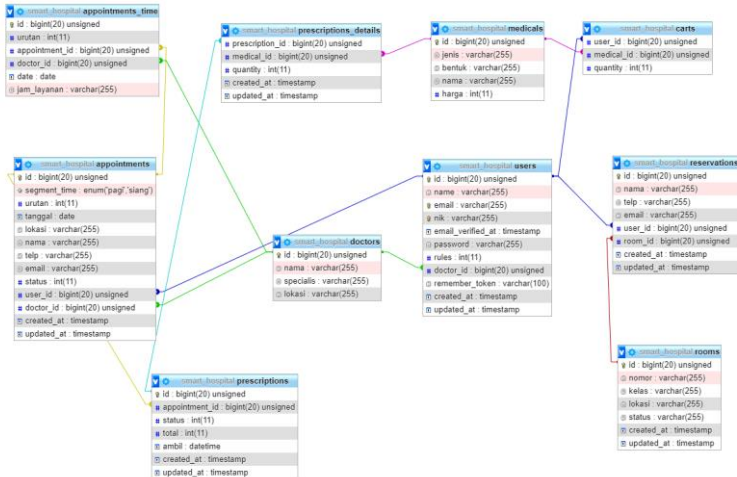
1. Schema::create('carts', function (Blueprint $table) {
2.     $table->bigInteger('user_id')->unsigned();
3.     $table->bigInteger('medical_id')->unsigned();
4.     $table->integer('quantity');
5.     $table->foreign('user_id')->references('id')->on('users');
6.     $table->foreign('medical_id')->references('id')->on('medicals');
7. });

```

Kode Sumber 4.10 MySQL carts

4.1.2 Hasil Query MySQL

Hasil dari implementasi *query* diatas akan menghasilkan skema database seperti pada Gambar 4.1.



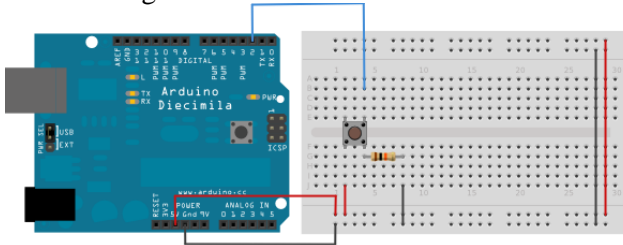
Gambar 4.1 Hasil Eksekusi MySQL

4.2 Implementasi Alat

Implementasi alat sistem Rumah Sakit Terintegrasi diperlukan beberapa rangkaian alat yang dihubungkan pada Wemos seperti penjelasan pada bab perancangan, Microcontroller akan dihubungkan dengan masing masing komponen melalui Relay 4 Channel 5V dengan mengirimkan arus listrik kepada relay, di mana microcontroller akan mengirimkan sinyal Low dan High untuk menghidupkan dan mematikan relay. Kemudian saat relay hidup saklar akan tertutup dan arus akan mengalir ke terminal NO(Normally Open).

4.2.1 Rangkaian Alat Rumah Sakit Terintegrasi

Seluruh rangkaian alat pada sistem Rumah Sakit Terintegrasi yang terhubung dengan relay akan menggunakan terminal NO. Rangkaian Sistem Rumah Sakit Terintegrasi adalah sebagai berikut:



Gambar 4.2 Rangkaian Sistem Ruumah Sakit Terintegrasi

PIN		Perangkat Terhubung	
4-Channel Relay 5V	Wemos D1 R1	Jenis Terminal	Nama Perangkat
GND	GND	-	-
IN 1	D2	NO	Button

Rangkaian sistem memiliki komponen-komponen yang terhubung pada *microcontroller* Wemos D1 R1, yaitu:

1. 4-Channel Relay 5V

Pada relay akan dihubungkan perangkat-perangkat pada kamar, yaitu *Solenoid Lock Door*, Lampu, Kipa. Perangkat akan dihubungkan pada relay melalui terminal NO(Normally Open) dan relay akan diberikan aliran listrik pada pin IN 1 pin D11 pada Wemos D1 R1 untuk menutup saklar pada relay dan mengaktifkan perangkat.

4.2.2 Implementasi Arduino

Rangkaian sistem dapat dijalankan dengan memasukkan kode program ke dalam Arduino melalui Arduino IDE. Implementasi fungsi-fungsi pada Arduino akan dijelaskan dan ditampilkan dalam bentuk Kode controller , code program lengkap akan dilampirkan sepenuhnya pada bagian Lampiran.

4.2.2.1 Library dan Konfigurasi WiFi

Bagian ini berisi semua *library* yang digunakan pada *microcontroller* serta konfigurasi untuk koneksi WiFi dan alamat host.

```
1. #include <SPI.h>
2. #include <MySQL_Connection.h>
3. #include <MySQL_Cursor.h>
4. #include <ESP8266WiFi.h>
5. #include <WiFiClient.h>
6. char ssid[] = "Fano";
7. char pass[] = "fano181407";
```

Kode Sumber 4.13 Library dan Konfigurasi WiFi

4.2.2.2 Konfigurasi Database

Bagian ini berisi semua konfigurasi untuk koneksi ke DB

```
1. char user[] = "user_ta";
2. char password[] = "user_ta";
3. WiFiClient client;
4. IPAddress server_addr(192, 168, 1, 2);
5. MySQL_Connection conn((Client *)&client);
6. MySQL_Cursor cur = MySQL_Cursor(&conn);
```

Kode Sumber 4.13 Library dan Konfigurasi WiFi

4.2.2.3 Deklarasi Pin

Bagian ini berisi deklarasi pin yang dipakai relay, dan konfigurasi untuk mendapatkan respon yang diinginkan dari web service.

```
1. const int in0 = D11; // Relay pin IN 1
2. const int in1 = D9; // Relay pin IN 2
3. const int in2 = D12; // Relay pin IN 3
4. const int in3 = D13; // Relay pin IN 4
```

Kode Sumber 4.9 Deklarasi Pin

4.2.2.4 Setup Arduino

Bagian ini adalah fungsi setup Arduino, fungsi ini akan dijalankan secara otomatis oleh Arduino ketika terhubung dengan arus listrik, fungsi ini akan menjalankan fungsi connectWifi yang berisi kode program untuk menghubungkan Arduino dengan *WiFi* dan *Web Server*.

```
1. void setup() {
2.   Serial.begin(9600);
3.   connectWifi(); // Fungsi connect WiFi
4.   pinMode(in0, OUTPUT); //Pin pintu
5.   pinMode(in1, OUTPUT); //Pin lampu
6.   pinMode(in2, OUTPUT); //Pin kipas
7.   pinMode(in3, OUTPUT); //Pin lift
8. }
```

Kode Sumber 4.10 Setup Arduino

```
1. void connectWifi(){
2.   // WiFi Connect
3.   Serial.print("Connecting to ");
4.   Serial.println(ssid);
5.   WiFi.begin(ssid, pass);
6.   while (WiFi.status() != WL_CONNECTED) {
```

```

7.      delay(200);
8.      Serial.print(".");
9.    }
10.   Serial.println("");
11.   Serial.println("WiFi Connected");
12.   Serial.print("Assigned IP: ");
13.   Serial.println(WiFi.localIP());
14.   //MySQL Connect
15.   Serial.println("Connecting to database");
16.   while (conn.connect(server_addr, 3306, user, pass
    word) != true) {
17.     delay(200);
18.     Serial.print ( "." );
19.   }
20.   Serial.println("Connected to SQL Server!");
21.   Serial.println("");
22. }

```

Kode Sumber 4.11 Fungsi Connect WiFi

4.2.2.5 Implementasi Loop Program

Setelah menjalankan fungsi void setup, Arduino akan melanjutkan nya dengan membaca fungsi void loop, fungsi ini akan terus melakukan pengulangan selama Arduino hidup, pada fungsi loop ini Arduino akan melakukan HTTP(Hypertext Transfer Protocol) *request* kepada *web server*, dan *web server* akan mengembalikan sebuah *response*. Arduino melakukan HTTP *request* akan menggunakan metode GET dengan parameter *id* dari Arduino yang mengirimkan *request*. Kemudian, *Web Server* akan mengembalikan sebuah *response* berisi status dari pintu, lampu, dan kipas.

```

1.  // ----- Function Loop
2.  void loop() {
3.    getData(); //HTTP Request data
4.    packData(); //Menyimpan response
5.    cekRuang(); //Buka atau Tutup pintu
6.  }

```


Kode Sumber 4.12 Fungsi Loop

```
function getData(){
  IF client melakukan request THEN
    Mencoba mendapatkan data;
  IF data ada
    return data
  ELSE
    return error
  END
}
```

Gambar 4.3 Pseudocode Fungsi Mengirim HTTP Request

Pseudocode fungsi `getData()` ini digunakan oleh client untuk mengirimkan *request* ke *web server*, jika server mengirimkan *response* dengan data yang diminta maka client akan mendapatkan data tersebut.

4.2.2.6 Implementasi Ruang Kamar Rumah Sakit

Bagian ini adalah implementasi dari kamar Rumah Sakit Terintegrasi yang menggunakan *Button*. Implementasi fungsi `room` ditampilkan dengan nama fungsi `cekPintu()` dalam bentuk *pseudocode* sebagai berikut.

```
function cekRuang (){
  IF status Rooms = 2 THEN
    Kosong
  ELSE
    Penuh
  END
}
```

Gambar 4.4 Pseudocode Membuka dan Menutup Pintu

Pseudocode fungsi `cekRuang()` ini digunakan untuk Monitoring pada website. Jika status pintu adalah 2 maka ruangan

kosong dan jika status pintu adalah 0, maka ruangan penuh. Kode lengkap program akan dilampirkan pada lampiran.

4.3 Implementasi Sistem berbasis Web

Pada sistem *monitoring* berbasis web ini akan memiliki beberapa fungsi yang digunakan untuk system registrasi periksa , reservasi rumah sakit, resep obat dan juga generate qrcode.

Web ini digunakan dengan *framework* Laravel 5.8 dan digunakan menggunakan aplikasi Visual Code Studio. Pertama yang harus dilakukan adalah konfigurasi *database* agar *framework* dapat terhubung dengan *database* penulis. Berikut adalah *Source Code* konfigurasi *database*

```

1. APP_NAME=Laravel
2. APP_ENV=local
3. APP_KEY=base64:zUTms48M/QgA9x9vhEF0w2yd0hbKLfkJ/0YW
  dVApAGI=
4. APP_DEBUG=true
5. APP_URL=http://localhost
6.
7. LOG_CHANNEL=stack
8.
9. DB_CONNECTION=mysql
10. DB_HOST=127.0.0.1
11. DB_PORT=3306
12. DB_DATABASE=smart_hospital
13. DB_USERNAME=root
14. DB_PASSWORD=
15.
16. BROADCAST_DRIVER=log
17. CACHE_DRIVER=file
18. QUEUE_CONNECTION=sync
19. SESSION_DRIVER=file
20. SESSION_LIFETIME=120
21.
22. REDIS_HOST=127.0.0.1

```

```

23. REDIS_PASSWORD=null
24. REDIS_PORT=6379
25.
26. MAIL_MAILER=smtp
27. MAIL_HOST=smtp.mailtrap.io
28. MAIL_PORT=2525
29. MAIL_USERNAME=null
30. MAIL_PASSWORD=null
31. MAIL_ENCRYPTION=null
32. MAIL_FROM_ADDRESS=null
33. MAIL_FROM_NAME="${APP_NAME}"
34.
35. AWS_ACCESS_KEY_ID=
36. AWS_SECRET_ACCESS_KEY=
37. AWS_DEFAULT_REGION=us-east-1
38. AWS_BUCKET=
39.
40. PUSHER_APP_ID=
41. PUSHER_APP_KEY=
42. PUSHER_APP_SECRET=
43. PUSHER_APP_CLUSTER=mt1
44.
45. MIX_PUSHER_APP_KEY="${PUSHER_APP_KEY}"
46. MIX_PUSHER_APP_CLUSTER="${PUSHER_APP_CLUSTER}"

```

Kode Sumber 4.17 Menampilkan Data Barang

Konfigurasi *database* berada pada folder *application/config/database.php*, *database* pada Laravel 5 dihubungkan dengan tabel yang sama dengan yang di akses oleh system Rumah Sakit Terintregasi .

Pada tahap ini, penulis menggunakan sepuluh *controller* dan sembilan *model*. Nantinya *controller* ini akan memiliki fungsi untuk menampilkan data-data barang dan pelanggan yang terdaftar di *database*. *Controller* dan *Model* yang ada pada sistem *Monitoring* adalah sebagai berikut:

1. DoctorController.php

Controller ini berfungsi untuk Mengambil data dokter.

2. **AppointmentController.php**

Controller ini digunakan untuk menyimpan data priksa , menggenerate qrcode dan Jam priksa

3. **RoomsController.php**

Model ini digunakan untuk menyimpan monitoring daftar kamar dan status kamar

4. **ReservationController.php**

Model ini digunakan oleh admin maupun user pada saat akan melihat list kamar , memesan kamar , dan juga acc request kamar yang di minta oleh user dan menggenerate qrcode

5. **MedicalController.php**

Controller ini berfungsi untuk mengambil data obat

6. **PrescriptionController.php**

Controller ini digunakan oleh admin untuk memberikan obat kepada pasien dan menggenerate qrcode

7. **PrescriptionDetailsController.php**

Controller ini berfungsi untuk menyimpan list nama obat dan jumlah yang diterima oleh pasien

8. **CartController.php**

Controller ini berfungsi menyimpan data obat sementara.

9. **ScanController.php**

Controller ini berfungsi Mengubah data setelah di post

4.3.1 Daftar Priksa

User membutuhkan fitur untuk dapat mendapatkan data dari *database*. Ini sangat diperlukan jika user akan mendaftar priksa. Kemudian akan digunakan *controller* barang pada web. Karna terlalu Panjang maka akan di tulis menggunakan pseudocode. Fungsi ini menggunakan Appointment.php

```
public function store(Request $request) {
    IF segmentime = pagi THEN
        cek_segment
    IF cek_segment
        POST
    ELSE

    IF segmentime = siang THEN
        cek_segment
    IF !cek_segment
    IF cek_segment
        POST
    ELSE
```

Kode Sumber 4.18 Mengupload Registrasi Priksa

Fungsi dibawah digunakan untuk Mengambil data list dan detail registrasi priksa

```
1. public function index()
2. {
3.     $appointments = Appointment::select()-
        >where('user_id', Auth::user()->id)->get();
4.     $appointmentstime = Appointmentsegment::all();
```

```

5.     return view('appointment', ['appointments' => $appointments, 'ap
pointmentstime' => $appointmentstime]);
6.     // return view('appointment')->with(compact('data'));
7. }
8.
9. public function show($id)
10. {
11.     $appointment = Appointment::find($id);
12.     return view('appointmentsingle', ['appointment' => $appointment
]);
13. }

```

Kode Sumber 4.19 Mendapatkan History Priksa

4.3.2 Monitoring Kamar

Digunakan Untuk monitoring Kamar secara Live di filter menggunakan status kamar

```

1. public function index()
2. {
3.     $rooms = Room::all();
4.     return view('room', ['rooms' => $rooms]);
5. }

```

4.3.3 Reservasi Kamar

Fungsi yang digunakan untuk melakukan reservasi request sebagai user dan reservasi sebagai admin dalam sistem *Rumah Sakit Terintregasi*.

```

1. public function show($id)
2. {
3.     $rooms = Room::find($id);
4.     return view('roomsingle', ['rooms' => $rooms]);
5. }

```

Kode Sumber 4.21 Form kamar user

Fungsi di atas merupakan fungsi yang digunakan untuk mengambil id kamar yang telah dipilih kemudian menggenerasinya menjadi form reservasi sesuai id

```

1.  public function store(Request $request)
2.  {
3.      Reservations::create([
4.          'nama' => $request->name,
5.          'telp' => $request->telp,
6.          'email' => $request->email,
7.          'user_id' => Auth::id(),
8.          'room_id' => $request->roomid,
9.          'status' => '1',
10.     ]);
11.     $room = Room::where('id','=', $request->roomid)->get();
12.     $room->status = '1';
13.     return redirect('/kamar');
14. }
```

Kode Sumber 4.22 Menyimpan data reservasi user

Fungsi di atas merupakan fungsi yang digunakan untuk menyimpan data Reservasi dan mengubah status kamar menjadi 1

```

1.  function reservation()
2.  {
3.      $room_list = Room::groupBy('lokasi')->get();
4.      $user_list = Appointment::where('doctor_id', Auth::user()-
5.          >doctor_id)->get();
6.      return view('roomReservation', ['room_list' => $room_list, 'user_list' => $user_list]);
7.  }
```

Kode Sumber 4.23 Form Kamar Admin

Fungsi di atas merupakan fungsi yang digunakan untuk mengambil id pasien yang telah diperiksa sebelumnya kemudian menggenerate nya form reservasi

```

1.  public function upload(Request $request)
2.  {
3.      Reservations::create([
4.          'nama' => $request->name,
5.          'telp' => $request->telp,
6.          'email' => $request->email,
7.          'user_id' => Auth::id(),
8.          'room_id' => $request->roomid,
9.          'status' => '0',
10.     ]);
11.     $room = Room::where('id','=', $request->roomid)->get();
12.     $room->status = '1';
13.     return redirect('/kamar');
14. }
```

Kode Sumber 4.24 Menyimpan data reservasi admin

Fungsi di atas merupakan fungsi yang digunakan untuk menyimpan data Reservasi dan mengubah status kamar menjadi 1 , perbedaan nya terletak pada status reservasi dimana status menjadi 0

```

1.  function roomAcc()
2.  {
3.      $reservations = Reservations::where('status','1')->get();
4.      return view('roomAcc')->with('reservations', $reservations);
5.  }
6.  public function update($id)
7.  {
8.      $reservations = Reservations::find($id);
9.      $reservations->status = '0';
10.
11.     return redirect()->back()->with('message', 'SUCCESS');
12. }
```

Kode Sumber 4.25 Acc reservasi user

Fungsi yang digunakan oleh admin untuk men acc reservasi request dari user *Rumah Sakit Terintregasi* setelah sebelum nya user mereservasi kamar rawat inap.

4.3.4 List Kamar Reserved

Fungsi yang digunakan oleh user untuk melihat reserved list dan mendapatkan bukti berupa qrcode

```

1. public function reserved()
2. {
3.     $reservations = Reservations::select()-
4.     >where('user_id', Auth::user()->id)->get();
5.     return view('roomreserved', ['reservations' => $reservations]);
6. }
7. public function reservedsingle($id)
8. {
9.     $reservation = Reservations::find($id);
10.    return view('roomreservedsingle', ['reservation' => $reservation]);
11. }

```

Kode Sumber 4.26 Mendapatkan History Reservasi

4.3.5 Pemberian Obat

Fungsi yang digunakan oleh admin untuk memberikan Obat kepada pasien yang dirasa membutuhkan obat untuk membantu penyembuhan nya

```

1. function preception()
2. {
3.     $medical_list = Medical::groupBy('jenis')->get();
4.     $users = Appointment::where('doctor_id', Auth::user()-
5.     >doctor_id)->get();
6.     return view('medicalPreception', compact('medical_list', 'users'));
7. }

```

Kode Sumber 4.27 Pemberian Obat

```

7. public function index()
8. {

```

```

9.     $prescriptions = Prescriptions::all();
10.    $users = User::all();
11.    return view('medical', ['prescription' => $prescriptions , 'users' =>
    $users]);
12.  }
13.
14.  public function show($id)
15.  {
16.    $prescriptions = Prescriptions::all();
17.    return view('medicalsingel', ['prescriptions' => $prescriptions]);
18.  }

```

Kode Sumber 4.28 Mendapatkan History pemberian obat

4.3.6 Carts

Fungsi yang digunakan oleh admin sebagai *database* sementara sebelum dimasukkan ke *database*

```

1.  public function userCart($id) {
2.    $carts = Cart::with('medical')->where('user_id', $id)->get();
3.    return $carts;
4.  }
5.
6.  public function addCart(Request $request) {
7.    $cart = Cart::where('medical_id', $request->id)-
    >where('user_id', $request->user_id)->first();
8.
9.    if ($cart) {
10.     $updateCart = DB::table('carts')
11.     ->where('medical_id', $request->id)
12.     ->where('user_id', $request->user_id)
13.     ->update(['quantity' => $cart->quantity + $request-
    >quantity]);
14.    return $updateCart;
15.  } else {
16.    $newCart = new Cart;
17.    $newCart->medical_id = $request->id;
18.    $newCart->user_id = $request->user_id;
19.    $newCart->quantity = $request->quantity;
20.    $newCart->save();
21.  }
22.  return true;

```

23. }

Kode Sumber 4.29 Menggunakan charts

```

1. public function deleteCart(Request $request) {
2.     $cart = Cart::where('medical_id', $request->id)-
       >where('quantity', $request->quantity)->delete();
3.
4.     if ($cart) {
5.         return true;
6.     }
7.     return false;
8. }
```

Kode Sumber 4.30 Mendelete charts

Fungsi yang bertujuan untuk mendelete 1 row pada table charts melalui webtoon

```

1. public function checkout(Request $request) {
2.     $appointment = Appointment::where('user_id', $request->user_id)-
       >get();
3.     $total = 0;
4.     $carts = Cart::with('medical')->where('user_id', $request->user_id)-
       >get();
5.     foreach ($carts as $cart) {
6.         $total+=$cart->quantity * $cart->medical->harga;
7.     }
8.     $prescriptions = new Prescriptions;
9.     $prescriptions->appointment_id = $appointment->id;
10.    $prescriptions->status = 0;
11.    $prescriptions->total = $total;
12.    $prescriptions->save();
13.    if (!$prescriptions) {
14.        return false;
15.    }
16.    foreach ($carts as $cart) {
17.        $detail = new PrescriptionDetails;
```

```

17.     $detail->prescription_id = $prescriptions->id;
18.     $detail->medical_id = $cart->medical_id;
19.     $detail->quantity = $cart->quantity;
20.     $detail->save();
21.   }
22.   $deleteCarts = Cart::where('user_id', $request->user_id)-
    >delete();
23. }

```

Kode Sumber 4.31 Migrating charts

Fungsi yang bertujuan untuk memindahkan tabel charts ke tabel preception karna memang charts hanya merupakan tabel sementara

4.3.8 Scan QR Code

Fungsi yang untuk melakukan scan qrcode sebagai konfirmasi apakah pasien tersebut sudah pendaftaran atau belum.

```

1.  public function updateapp($id)
2.  {
3.    $appointments = Appointment::find($id);
4.    $appointments->status = '0';
5.
6.    return redirect()->back()->with('message', 'SUCCESS');
7.  }

```

Kode Sumber 4.32 QR code Appointment

```

1.  public function updatereserv($id)
2.  {
3.    $reservations = Reservations::find($id);
4.    $reservations->status = '0';
5.
6.    return redirect()->back()->with('message', 'SUCCESS');

```

```
7. }
```

Kode Sumber 4.33 QR code Appointment

```
1. public function updateobat($id)
2. {
3.     $prescription = Prescriptions::find($id);
4.     $prescription->status = '0';
5.
6.     return redirect()->back()->with('message', 'SUCCESS');
7. }
```

Kode Sumber 4.34 QR code Reservation

BAB V

UJICoba DAN EVALUASI

Pada bab ini dijelaskan tentang uji coba dan evaluasi dari implementasi yang telah dilakukan pada Tugas Akhir ini.

5.1 Lingkup Uji Coba

Pada lingkup uji coba, penulis menggunakan konektivitas jaringan rumah “*Home Network*” untuk melakukan uji coba Sistem Rumah Sakit Terintregasi, sistem monitor dan kontrol dijalankan pada laptop dengan spesifikasi sebagai berikut:

Tabel 0.1 Spesifikasi

Komponen	Spesifikasi
CPU	Intel(R) Core(TM) i7-6700HQ CPU @ 2.60GHz (8 CPUs), ~2.6GHz
Sistem Operasi	Windows 10 Pro N 64-bit (10.0, Build 18363)
Memori	8192MB RAM
Bios	BIOS Date: 04/22/16 14:05:07 Ver: 05.0000B
Penyimpanan	1 TB

Semua perangkat sudah terhubung dengan koneksi WiFi. Uji coba dilakukan menggunakan contoh kasus yang menggunakan system Rumah Sakit Terintregasi.

5.2 Sistem Rumah Sakit Terintregasi

Pengujian Sistem Rumah Sakit Terintregasi dilakukan untuk mengetahui hasil dari implementasi, dimana pengujian dilakukan dalam bentuk kasus yang telah dibuat.

1. Mendaftar Periksa

Langkah pertama Registrasi priksa dalam sistem Rumah Sakit Terintegrasi yang di dilakukan oleh user adalah mengisi form

17:57 192.168.1.2:8000/regist

Registrasi Priksa

Pilih Hari

Pilih spesialis

Pilih Lokasi

Pilih Dokter

Pilih Jam

Telp

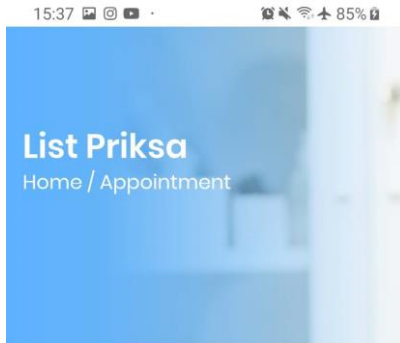
email

nama

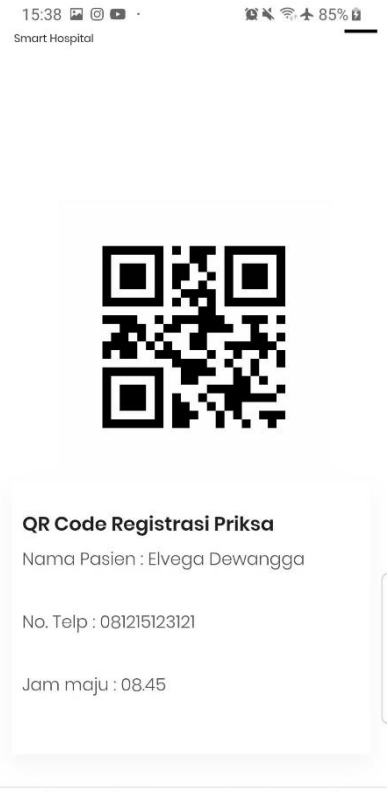
Confirm

Gambar 0.1 Tampilan Registrasi User

Setelah menekan tombol “*Confirm*”, form akan dimasukkan ke dimasukkan ke *database*.



Gambar 0.2 Tampilan List Registrasi

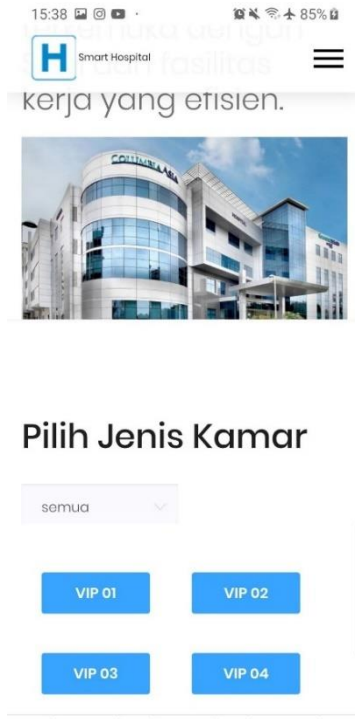


Gambar 0.3 Barang berhasil ditambah

Setelah Registrasi Priksa dilakukan, detail registras akan muncul di halaman list registrasi priksa dan memunculkan perkiraan Jam. QR code juga akan otomatis terbuat sesuai dengan ID Registrasi.

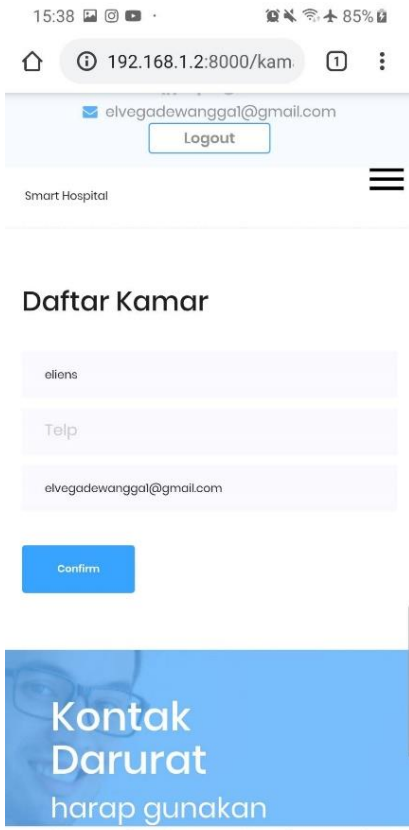
2. Reservasi kamar

Langkah pertama Reservasi Kamar dalam sistem Rumah Sakit Terintegrasi yang dilakukan oleh user adalah memilih kamar



Gambar 0.4 Halaman List kamar

Setelah menekan salah satu kamar user akan dibawa ke form reservasi kamar



15:38 192.168.1.2:8000/kam. 85%

Smart Hospital

Daftar Kamar

Nome

Telp

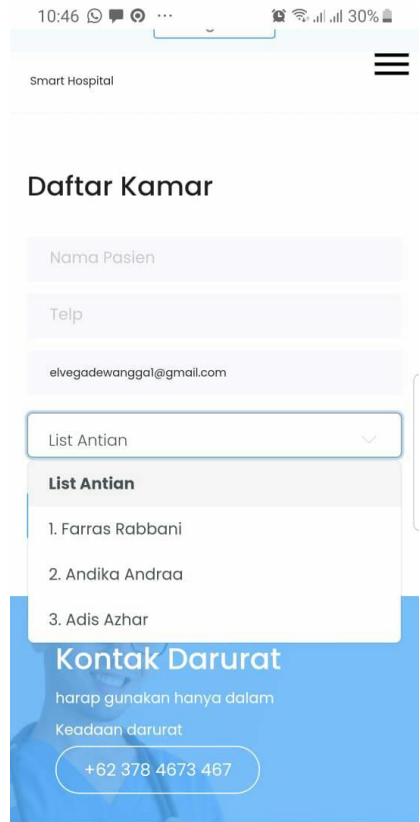
elvegadewanggal@gmail.com

Logout

Confirm

Kontak Darurat
harap gunakan

Gambar 0.5 Registrasi Kamar tanpa antrian



10:46 30%

Smart Hospital

Daftar Kamar

Nama Pasien

Telp

elvegadewanggal@gmail.com

List Antrian

List Antrian

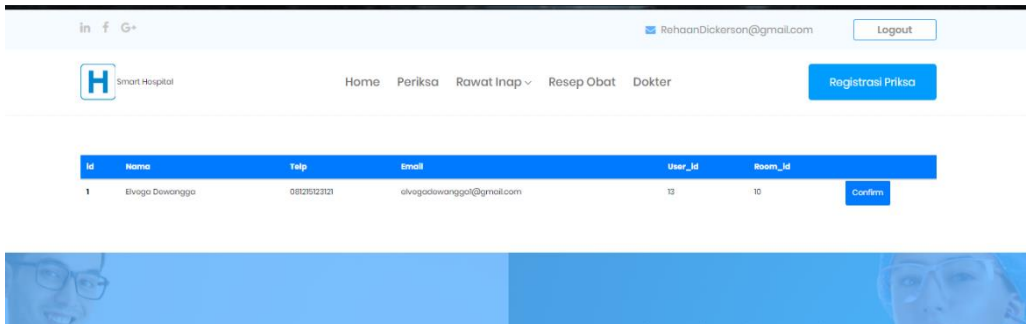
- Farras Rabbani
- Andika Andraa
- Adis Azhar

Kontak Darurat
harap gunakan hanya dalam keadaan darurat

+62 378 4673 467

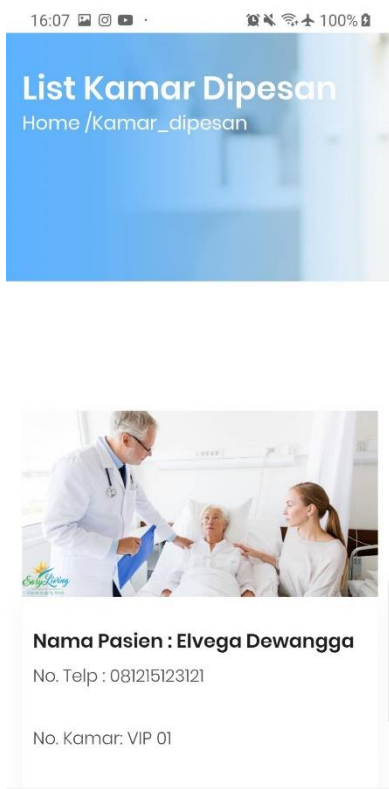
Gambar 0.6 Registrasi Kamar Dengan antrian

Setelah menekan tombol “*Confirm*”, form Reservasi kamar akan dimasukkan ke *database* dan harus menunggu admin untuk men acc *request* nya.



Gambar 0.7 Halaman acc kamar

Alur dilanjutkan dengan acc kamar oleh admin, dimana admin dapat mengkonfirmasi apakah pasien tersebut diperbolehkan menjalani rawat inap atau tidak



Gambar 0.8 Tampilan List Registrasi

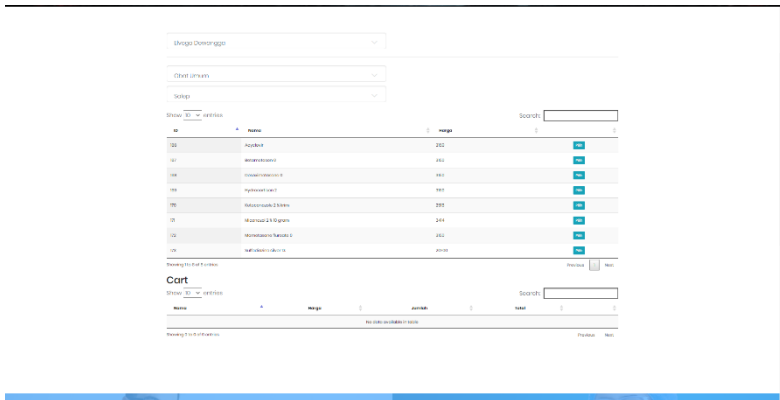


Gambar 0.9 Barang berhasil ditambah

Setelah Reservasi Kamar dilakukan, detail Reservasi akan muncul di halaman list Reservasi kamar dan QR code juga akan otomatis terbuat sesuai dengan ID Registrasi.

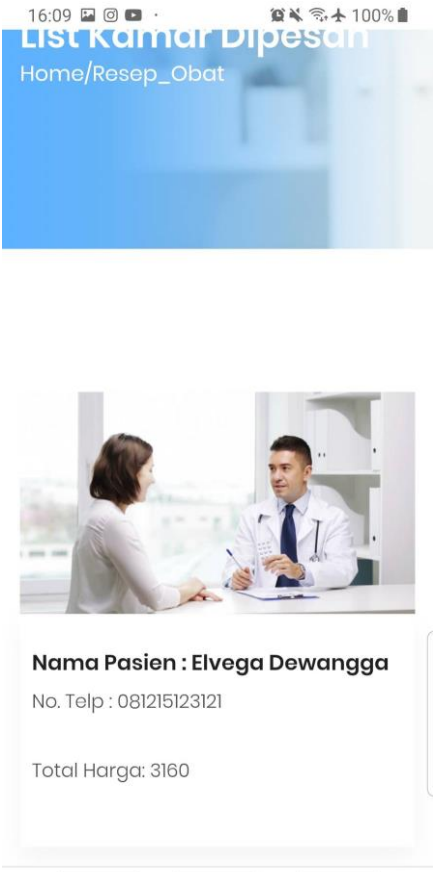
3. Pemberian resep obat

Langkah pertama Pemberian resep obat dalam sistem Rumah Sakit Terintegrasi yang di dilakukan oleh admin adalah memilih obat



Gambar 0.10 Halaman List kamar

Setelah memilih obat yang dibutuhkan , list obat akan masuk kedalam database dan akan muncul di page obat user



Gambar 0.11 Halaman List Obat

[Halaman ini sengaja dikosongkan]

BAB VI

KESIMPULAN

Pada bab ini akan dijelaskan kesimpulan dari hasil ujicoba yang telah dilakukan.

6.1 Kesimpulan

Kesimpulan yang diperoleh dari Tugas Akhir ini didasarkan pada hasil uji coba dan evaluasi. Kesimpulan Tugas Akhir ini adalah sebagai berikut:

1. Website Registrasi Periksa secara online , dimana pasien dapat beristirahat di rumah dan hanya datang ke rumah sakit saat giliran periksanya dengan perkiraan waktu priksa
2. Monitoring online kamar rumah sakit secara realtime dan memberikan fitur reservasi kamar
3. Memberikan resep obat secara online sehingga apoteker dapat menyiapkan obatnya terlebih dahulu
4. Scan Qrcode menggunakan aplikasi android yang kemudian di sinkronkan dengan web server
5. Sistem transaksi sudah berjalan sesuai keinginan.
6. Alat yang terhubung dengan *microcontroller* sudah bisa berfungsi dengan baik

6.2 Saran

Saran yang dapat diberikan dari hasil uji coba dan evaluasi adalah sebagai berikut:

1. Sistem untuk pasien di implementasikan menggunakan android
2. Memberikan Fitur deteksi orang di tiap kamar
3. Memberikan sensor suhu untuk pasien
- 4.

[Halaman ini sengaja dikosongkan]

DAFTAR PUSTAKA

1. (Arduino, 2018; Laravel, 2019; Lengstorf, 2010; Shah, 2020; Steyer & Steyer, 2018; Wheat & Wheat, 2011)Arduino. (2018). Arduino Uno Rev3. *Store.Arduino.Cc*.
2. Laravel. (2019). *Laravel - The PHP Framework For Web Artisans*. LARAVEL.
3. Lengstorf, J. (2010). PHP jQuery. In *Structure*.
<https://doi.org/10.1007/978-1-4302-2848-6>
4. Shah, C. (2020). MySQL. In *A Hands-On Introduction to Data Science*.
<https://doi.org/10.1017/9781108560412.008>
5. Steyer, R., & Steyer, R. (2018). jQuery. In *jQuery*.
<https://doi.org/10.3139/9783446456518.fm>
6. Wheat, D., & Wheat, D. (2011). Arduino Software. In *Arduino Internals*. https://doi.org/10.1007/978-1-4302-3883-6_5

LAMPIRAN

1. <https://github.com/EDewangga/SmartHospital>

BIODATA PENULIS



Elvega Dewangga lahir pada 18 April 1998 di Semarang. Penulis merupakan anak pertama dari tiga bersaudara. Hobi dari penulis adalah bermain game dan bermain futsal. Penulis memulai menempuh pendidikan di SD Haja Isriati Semarang (2004 – 2010). Kemudian pendidikan penulis berlanjut ke SMPN 5 Semarang (2010 – 2013). Setelah menyelesaikan pendidikan di SMPN 5 Semarang, penulis melanjutkan pendidikan ke SMA N 1 Semarang (2013 – 2016). Setelah lulus, penulis melanjutkan pendidikan ke jenjang yang lebih tinggi untuk mendapat gelar sarjana di Institut Teknologi Sepuluh Nopember (2016 - sekarang) dan mengambil jurusan Teknik Informatika. Penulis memilih untuk mengambil Rumpun Mata kuliah Arsitektur dan Jaringan Komputer (AJK). Penulis pernah menjadi bagian dari Himpunan Mahasiswa Teknik Computer-Informatika (HMTIC) sebanyak 2 periode (2017/2018 sebagai staff Hubungan Luar dan 2018/2019 sebagai Wakil Kepala Departemen Minat dan Bakat). Dan juga pernah ikut bagian saat Schematics berjalan (2017/2018 sebagai staff Humas dan 2018/2019 sebagai staff ahli REEVA). Penulis dapat dihubungi melalui email *elvegadewangga1@gmail.com*